Protein Science (1994), *3*:557–566. Cambridge University Press. Printed in the USA.

Analysis of protein transmembrane helical regions by a neural network

GEORGE W. DOMBI[1] AND JEANNETTE LAWRENCE[2]

[1] Surgery Department, Wayne State University, Detroit, Michigan 48201
[2] California Scientific Software, Nevada City, California 95959

(RECEIVED August 25, 1993; ACCEPTED February 3, 1994)

Abstract

Neural networks were used to generalize common themes found in transmembrane-spanning protein helices. Various-sized databases were used containing nonoverlapping sequences, each 25 amino acids long. Training consisted of sorting these sequences into 1 of 2 groups: transmembrane helical peptides or nontransmembrane peptides. Learning was measured using a test set 10% the size of the training set. As training set size increased from 214 sequences to 1,751 sequences, learning increased in a nonlinear manner from 75% to a high of 98%, then declined to a low of 87%. The final training database consisted of roughly equal numbers of transmembrane (928) and nontransmembrane (1,018) sequences. All transmembrane sequences were entered into the database with respect to their lipid membrane orientation: from inside the membrane to outside. Generalized transmembrane helix and nontransmembrane peptides were constructed from the maximally weighted connecting strengths of fully trained networks. Four generalized transmembrane helices were found to contain 9 consensus residues: a K-R-F triplet was found at the inside lipid interface, 2 isoleucine and 2 other phenylalanine residues were present in the helical body, and 2 tryptophan residues were found near the outside lipid interface. As a test of the training method, bacteriorhodopsin was examined to determine the position of its 7 transmembrane helices.

Keywords: computer analysis of protein structure; neural networks; transmembrane protein helix

The goal of much of modern protein science is to solve the problem of the second code—to determine the folded structure (tertiary structure) of a functional protein based on primary structure information of its constituent amino acids. Even though the tertiary structure of a protein can be complex, it is composed of peptide units held together by repeating patterns of hydrogen bonds called secondary structure (Sasagawa & Tajima, 1993). Continuous sequences of amino acids are broadly classified into various secondary structures including α-helix, β-sheet, and random-coil regions. A sure sign of progress toward the goal of tertiary structure prediction would be reliable determination of secondary structure from amino acid sequence information (Bohr et al., 1988).

One of the newer techniques to be applied to problems of protein structure is that of neural networks—computer programs that can detect patterns and correlations in data by learning to place increasing weight on critical information and reducing or ignoring other information (Hirst & Sternberg, 1992). Neural networks are well suited to the problem of gleaning structural information from local amino acid patterns. Neural networks can be trained on examples of amino acid sequences with known secondary structures, then tested with a different set of known sequences for evaluation of test accuracy. A number of investigators have already applied neural networks to cytosolic proteins (Qian & Sejnowski, 1988; Holley & Karplus, 1989; Kneller et al., 1990; Hayward & Collins, 1992) and also to 1 transmembrane protein, rhodopsin (Bohr et al., 1988).

Even though secondary-structure determination is the penultimate goal of primary structure analysis, there is information to be learned from analysis of secondary structure for its own sake. Our interest is in expanding the use of neural network techniques for secondary-structure analysis by a more thorough examination of the helical regions in transmembrane proteins. To this end, we created a sorting task for a neural network that required the correct classification of equal length peptides to either of 2 groups: transmembrane helical peptides or nontransmembrane peptides. In the process of learning to make these classifications correctly, the network was forced to generalize from the training examples and set up a matrix of internal weights that represented those amino acids important to the structure of the transmembrane helix.

We will report our results of this classification task and show evidence that a commercially available, desktop computer-based neural network program can successfully discern general features of the transmembrane helical regions and can also distinguish

Reprint requests to: George W. Dombi, Wayne State University, 1221 Elliman Building, 421 East Canfield Avenue, Detroit, Michigan 48201.

these features from globular peptide segments of equal length. Learning curves will be presented showing the effects of training set size, addition of data noise, and placement of the helical region in the training set sequence. Summary tables will also be presented showing the effects of these variables on test set sorting accuracy. A table showing false-positive and false-negative test set assignments will be presented. A Hinton diagram will be presented to show the relative strengths of neural weight connections from the weight matrix of a fully trained network. Examples will be presented of "generalized" transmembrane sequences and "generalized" nontransmembrane sequences based on these neural weight connections from 4 fully trained networks. Finally, a complete protein, bacteriorhodopsin, will be searched for correct placement of its known transmembrane helices.

Results and discussion

Neural network software

The use of neural networks to recognize complex patterns is a potential tool for the protein scientist. The ability to use a neural network is made easier by the commercial availability of programs such as *BrainMaker*, which run on an IBM-PC type computer. We ran this program on 2 different computers. Our early neural networks, with databases less than 500 sequences, were trained in about 2 h on a 286 computer with a math coprocessor. Later neural networks, with databases containing 500–1,750 sequences, were trained in 1–2 h on a 386 computer with a math coprocessor. We found it convenient that this particular neural network-generating program, *BrainMaker*, allowed for the representation of a peptide sequence as a series of letters and numbers (e.g., A1 T2 K3 ... G25). This representation was more easily understood than the commonly used but less obvious method of representing the data as a sequence of binary 1's and 0's (Bohr et al., 1988; Holley & Karplus, 1989; Ladunga et al., 1991).

In order to accommodate a peptide length of 25, we designed all neural networks with 500 input neurons (20 amino acids × 25 positions) plus 1 threshold input neuron, 1 hidden neuron and 1 hidden threshold neuron, and 2 output neurons, giving a total of 505 neural connections. Using 500 input neurons permitted the amino acid window to be long enough to span the length of the transmembrane region (18–20 amino acids) while maintaining a number below the 512 input neuron limit imposed by *Brainmaker* standard version software. The decision to design neural networks with a single neuron in the hidden layer was made so as to allow for the construction of the generalized transmembrane and generalized nontransmembrane sequences from the neural weight matrix as explained below. Also, our choice of 1 hidden neuron was reasonable compared to previous neural net models that have used either 2 hidden neurons (Holley & Karplus, 1989), or 0 hidden neurons—a Perceptron model that directly connected the input layer to 2 output neurons (Kneller et al., 1990).

Biphasic nature of neural network learning

A learning curve was generated for each of 12 individual neural networks by plotting the number of correctly assigned training facts as a function of the number of iterative cycles through the training set. Figure 1 shows learning curves from 10 different neural networks trained on various-sized training sets. It can be seen that the learning curves had both a steep initial portion and a more gradual tail.

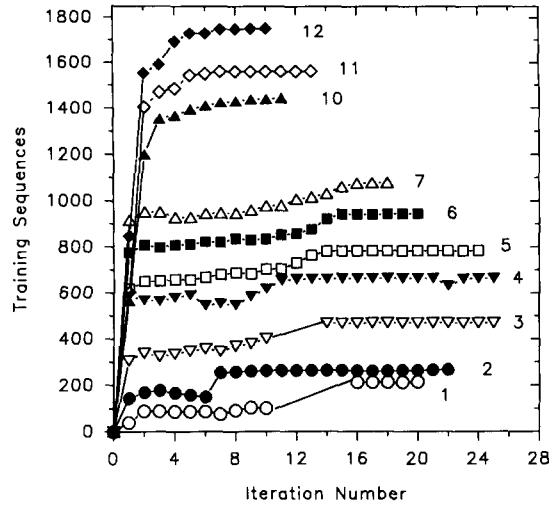

Fig. 1. Neural network learning curves. The biphasic nature of the learning curve becomes more obvious as the size of the training set increases. The numbers 1–7 and 10–12 identify the corresponding neural networks as described in Table 1.

This biphasic nature of learning was more obvious as the size of the training set was increased (Table 1). In all but neural network 1, assignment of peptide sequences in the training set was correct more than 50% of the time after the first cycle, which would be expected in a network with only 2 possible outputs. But the level of 90% correct training was reached in less than half of the total number of iterations, which indicated a nonlinear learning curve. By way of comparison with a linear learning curve starting at same level of 50% correct training, the 90%

Table 1. *Summary statistics of neural network learning based on the size of the training set*

Network number	Training set size	Iterations at 50%	Iterations at 90%	Iterations at 100%	Test set size	Percent correct
1	214	10–11	~15	20	24	75
2	266	0–1	7–8	23	30	80
3	471	0–1	11–12	32	52	83
4	664	0–1	9–10	30	74	92
5	785	0–1	11–12	24	87	94
6	945	0–1	9–10	20	105	98
7	1,078	0–1	9–10	18	120	95
10	1,445	0–1	2–3	11	161	96
11[a]	1,563	1–2	2–3	13	174	96
12[b]	1,751	1–2	2–3	10	195	87

[a] Network 11 used a training set with the lipid interface set between the third and fourth amino acids in the 25-residue sequences. Networks 1–10 used training sets with the lipid interface set between the seventh and eighth amino acids.

[b] Network 12 used a training set with the first amino acid of the 25-residue sequence set at the third position inside the lipid interface.

level would be expected to be reached after 80% of the total number of iterations.

Effects of training set size on test set sorting accuracy

Training set size had a nonlinear effect on test set sorting accuracy (Table 1). Network 1 was 75% accurate. This network was trained with 214 protein sequences, which was roughly half the number of the 500 input neural connections. Test set sorting accuracy increased as a function of training set size until it reached a maximum of 98% with network 6, which used a training set about twice the size of the 500 input neural connections. Further, there was no real increase in the level of sorting accuracy (96%) with network 11, which used a training set roughly equal to 3× the number of input neural connections.

Previously reported neural network studies have been concerned with prediction of secondary structure from primary structure information. Our goal is a more limited one: train a neural network to distinguish a whole transmembrane helix, as a unit, from globular peptides of the same size. This is a sorting problem. Nevertheless, it is useful to compare our networks to those other studies as benchmarks. Based on those previously reported studies, the number of training set sequences per neural connection has been shown to play an important role in network learning. In the Bohr study (Bohr et al., 1988), a neural network was created with about 40,000 total neural connections (1,020 input, 40 hidden, and 2 output neurons) and was trained on a database containing about 15,000 peptide sequences. This resulted in a prediction accuracy of about 73% for the amino acids comprising the transmembrane helices in rhodopsin. In that case, there were 2–3× more neural connections than training set sequences. This overabundance of neural connections reduces the need to abstract information from the training set and can lead to possible memorization of the training set resulting in a limited ability to predict a nonhomologous testing set. In the Kneller study (Kneller et al., 1990), a neural network was created with 819 neural connections (273 input, 0 hidden, and 3 output neurons) and was trained with a database of about 20,000 peptide sequences. This resulted in an improved prediction accuracy of about 79% for amino acids comprising the helical regions in proteins of the all-α-helix class. In our study we created a network with 505 neural connections (500 input, 1 hidden, and 2 output neurons) and trained it with a database of 1,563 peptide sequences in network 11. This resulted in a test set sorting accuracy of about 96%. In our case, having more than 3× the number of training sequences than neural connections forced the network to abstract information from the training set. This allowed a better chance of high test set sorting accuracy with a nonhomologous testing set and a high chance of test set sorting accuracy with a homologous test set.

Effects of sequence homology on test set sorting accuracy

Our best test set sorting accuracy was between 95 and 98%, which probably reflects the high homology between proteins in our training and test data sets. Reports from other workers using both ternary and binary models have suggested that there may be a limit to the degree of discernment possible when neural networks are used for predicting protein structure. The limits of prediction accuracy tend to increase as a function of homology between the training and test data sets. The ternary problem is to distinguish between amino acid sequences that comprise α-helices, β-sheets, or random coils. Results of neural networks applied to that problem have been reported between 63 and 64% (Qian & Sejnowski, 1988; Holley & Karplus, 1989) and a limit of 65% has been suggested (see review by Hirst & Sternberg, 1992). For binary models such as predicting α-helix vs. non-α-helix (Hayward & Collins, 1992) or β-turns vs. non-β-turns (McGregor et al., 1989), there has been proposed a theoretical limit of 73–78%. The improvement of the binary model over the ternary model can be ascribed purely to the reduction in the number of predictable classes (Kneller et al., 1990). These theoretical limits were determined using a minimum of homologous sequences in the training and test data sets. When homology was increased, prediction accuracy increased. In a ternary model, the prediction accuracy rose to 79% when using a class of proteins that contained nearly all α-helical structure (Kneller et al., 1990). Based on the proportions from these theoretical limits, we calculated a new theoretical limit of 89–95% for the prediction accuracy of a binary model when there is a similarly high degree of homology. The test set sorting accuracy of our binary model (95–98%) falls just at and beyond the upper level of this new theoretical limit, which may mean our level of homology is even higher than that used in Kneller's study.

Effects of lipid membrane placement in the training set

Placement of the lipid interface caused no effect on the learning curve. There was no difference when training set sequences were trimmed so that the inside lipid interface fell between the seventh and eighth amino acid with no outside lipid interface (network 10), or between the third and fourth position including a small part of the outside lipid interface (network 11), or when the inside lipid interface was absent and a 7-member tail was included from the outside lipid interface (network 12). Table 1 shows similar training summary statistics for networks 10, 11, and 12, and Figure 1 shows similar learning curves for these 3 networks as well. On the other hand, there was a reduction in test set sorting accuracy when the inside lipid interface was removed from the training set sequences. The test set sorting accuracy fell from 96% (networks 10 and 11) to 87% (network 12). This is taken to mean that there was a loss of important structural information when the inside lipid interface was removed that was not compensated by the inclusion of amino acids from the outside lipid interface. This may support the idea that insertion of the helix is from the inside and the major amino acid sequence information for stopping insertion is on the inside lipid interface.

Training with added noise

Because we used a symbolic rather than a numeric representation of the amino acid sequences, an individual input neuron had only 2 possible values. It was either on or off, e.g., the neuron representing alanine at position 22, neuron A22, had a discrete value of either 0 or 1. In an attempt to give input neurons a more continuous scale, ±5% noise was added to the training data set. Addition of noise is a built-in option with *BrainMaker* software. We asked the question whether addition of noise to the data would in some way enhance learning rate or test set sorting accuracy of a neural network. Figure 2 shows the comparison of neural networks 8 and 9. Both were trained to 100% with the

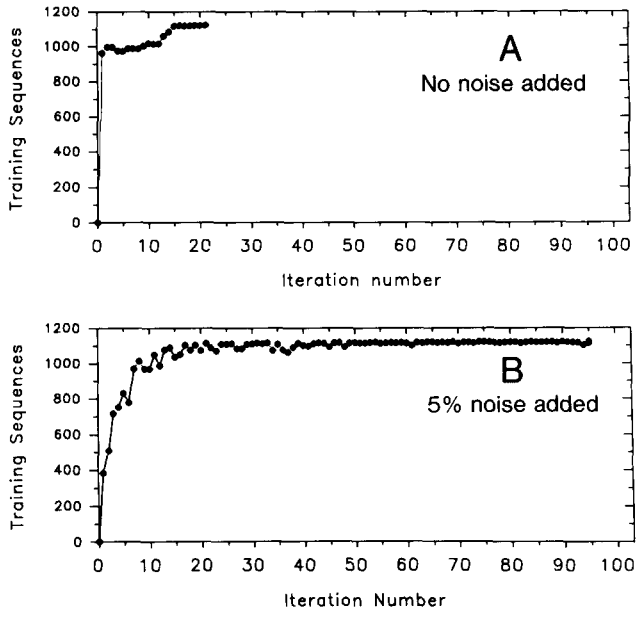

Fig. 2. Effect of adding ±5% noise to the training set. No noise was added to network 8 (**A**), but noise was added to network 9 (**B**). An increase in the time to 100% training was seen in the learning curves of network 9. Both networks used the same initial training set with 1,123 peptide sequences.

same training set containing 1,123 peptide sequences. It can readily be seen that training with added noise took about 4.5× longer than without added noise. Comparison of the 2 training methods indicated no improvement in test set sorting accuracy (Table 2). Network 9 was trained with added noise and correctly assigned 91% of the test set. This was below the 94% level of network 8, which used the same training set but with no added noise.

Other factors in network learning

In addition to the use of highly homologous proteins in our database, our high test set sorting accuracy may also have been the result of 2 other factors. The first factor to consider is the method of database analysis. In the previously reported neural network studies, sliding windows of different sizes were used to scan the databases: 13 residues (Qian & Sejnowski, 1988; Kneller et al., 1990), 17 residues (Holley & Karplus, 1989), or 51 residues (Bohr et al., 1988). All of the amino acids in a window were used to train the network to assign the secondary structure of the central amino acid. In our study, preset slices of 25 residues were taken as a unit and assigned to either the transmembrane helical group or not. We did not consider each individual amino acid. Our study also used no sliding or overlapping sequences. Each consecutive sequence of 25 residues was judged as a separate unit to be a transmembrane helix or not.

The second factor may have been the reading frame direction. Although not explicitly stated in the previous studies, it is assumed that all of the sequences were read in the same direction, from N-terminal to C-terminal. In our study, the helical regions were aligned from inside to outside the membrane. This alignment may have increased the amount of information built into the training data set and made test set sorting accuracy higher.

Table 2. *Summary statistics of neural network learning based on the addition of noise to the training set*

Network number	Training set size	Iterations at 50%	Iterations at 90%	Iterations at 100%	Test set size	Percent correct
8[a]	1,123	0-1	9-10	21	125	94
9[b]	1,123	2-3	7-8	95	125	91

[a] No added noise.
[b] Five percent added noise in the training set.

Testing a fully trained network

No network reported here reached a test set sorting accuracy of 100%. It is useful to examine those sequences that were falsely sorted. Table 3 presents the false-positive and false-negative assignments for neural network 11, which had a 96% test set sorting accuracy. There were 2 mislabeled transmembrane helical sequences (1-2) and 5 mislabeled nontransmembrane sequences (3-7). The value of the helix to non-helix ratio, shown in Table 3, was a relative measure of how closely a given test sequence resembled the important weights determined by the network during the training session. These ratio values consisted of a scale from 0 to 8 and were rounded off to the nearest whole number. A sequence that fully matched the important transmembrane helical training weights had a ratio of 8/0, whereas a sequence that fully matched the important nontransmembrane training weights had a ratio of 0/8. The mislabeled sequences failed to match the proper training weights. Sequence 7 came from the precursor peptide region of human tumor necrosis factor. In general, the precursor peptide region is similar to a transmembrane helix but is usually longer than 18-20 amino acids. Other precursor peptide regions were included in the database, but none of those were falsely labeled, as was sequence 7.

Weight table matrix of trained network

A fully trained neural network is characterized by the neuron weight matrix that forms as a result of training. An example of a weight matrix is presented as a Hinton diagram for network 11 in Figure 3, where weights range in size from −3 to 3. The matrix gives the relative importance of each of the 20 amino acids at each position of the 25 residue peptides. Open boxes represent positive weights that favor classifying the sequence as a transmembrane helix structure, whereas closed boxes represent negative weights that support classifying the sequence as a nontransmembrane structure.

Generalized sequences from weight table matrix

Because all neural networks in this study used a single hidden neuron, it was possible to make direct comparisons of the neuron weights for each amino acid at any given position in the 25-residue training sequence. By taking the amino acid at each position with the largest positive and largest negative value, we constructed the neural network's "generalized" transmembrane and "generalized" nontransmembrane peptides (see Table 4). For

Table 3. *Listing of false-positive and false-negative sequence assignments*[a]

No.	Protein[b]	Sequence structure (25 aa)	True assignment	Ratio
1	Cytochrome oxidase subunit I, *Escherichia coli*	AVPVYYDVDEDFSKVIWTIIMGAFG	Helix	0/7
2	Cytochrome oxidase subunit III, yeast	TAGHHVGYETTIIYTHVLDVIWLFL	Helix	6/2
3	Alcohol dehydrogenase, rabbit liver	VIGRLDTVVAALLSCHGACGTSVIV	Nonhelix	5/3
4	Glutamate dehydrogenase, bovine, DEBOE	ADKIFLERNIMVIPDLYLNAGGVIV	Nonhelix	4/4
5	Granulocyte colony-stimulating factor receptor, mouse	TQAFLFCLVPWEDSVQLLDQAELHA	Nonhelix	2/6
6	Lactate dehydrogenase, *Bacillus stearothermophilus*, DEBSLF	QIIEKKGATYYGIAMGLARVTRAIL	Nonhelix	8/0
7[c]	Tumor necrosis factor receptor, human preprotein	LLPLVLLELLVGIYPSGVIGLVPHL	Nonhelix	8/0

[a] False positives were nonhelical sequences classified by the neural network as belonging to the transmembrane helical group. False negatives were helical sequences misclassified as nonhelical. Results from the test data set for neural network 11 are shown below where 7 of the 174 test sequences were incorrectly labeled. Correct structure assignment is listed as well as the determined ratio of helix/nonhelix weights, upon which the incorrect assignment was made. Expected ratio for a correctly assigned transmembrane helix equals 8/0; expected ratio for a correct nontransmembrane helix equals 0/8.

[b] All sequences are listed as used in the database by neural network 11: from N-terminal to C-terminal including no. 1, which is presented in the reverse direction from C-terminal to N-terminal.

[c] Sequence for no. 7 is part of the preprotein sequence.

this purpose, 4 of the later networks were used because they were trained with sufficient numbers of amino acid sequences to merit good generalization of training set information. There were between 2 and 3× the number of training sequences compared to the number of input neurons in these networks.

Generalized transmembrane helices

For the sake of comparison, the generalized sequences in Table 4 were aligned at the lipid membrane. Generalized sequences from networks 8 and 10 had a 7-member tail of amino acids on the inside of the lipid membrane and no amino acids on the outside membrane interface. The generalized sequence from network 11 had a 3-member tail on the inside and a 2-member tail on the outside lipid interface. The generalized sequence from network 12 had no amino acids on the inside and a 7-member tail on the outside of the lipid membrane. The generalized transmembrane sequences contained a polar region near the inside lipid interface, a hydrophobic sequence running the length of the lipid bilayer, and a terminal region of identical amino acids. The amphipathic nature of the helix with a charged lipid interface region and a hydrophobic body was the expected result as described by others (Rao & Argos, 1986; Hartmann et al., 1989; Jones et al., 1992; von Heijne, 1992).

The inside lipid boundary region of the generalized transmembrane helices contained a triplet sequence, K-R-F, that spanned the lipid membrane. The body of the generalized transmembrane helices contained a phenylalanine immediately following the K-R doublet as the first residue inside the lipid layer and further consensus hydrophobic residues of 2 isoleucines and 2 other phenylalanine residues. The presence of these residues supports the observation that leucine, isoleucine, phenylalanine, and valine make up about 50% of the residues of transmembrane proteins (von Heijne, 1981). This is not to say that all transmembrane helices contain the same features as the generalized transmembrane helices, but that these features were landmarks for the neural network analyses.

The idea that phenylalanine supports the formation of helical structure has also been shown in the study of protein folding patterns predicted by neural networks using amino acid composition data (Dubchak et al., 1993). That study also indicated that both tryptophan and lysine disfavored predictions of the protein class containing the folding pattern known as 4-α-helical bundles.

Near the outside lipid interface, the generalized transmembrane helices were found to contain tryptophan. This residue has been previously reported as often found at the boundary region of the lipid interface (Reithmeier & Deber, 1991). Tryptophan appears to cap the outside end of the helix and may provide a role in helix orientation when it is initially inserted into the membrane. Likewise, the presence of the K-R doublet may provide a strong stop message for the helix insertion process.

The generalized transmembrane helices also seem to support the "positive inside rule" of more positive charged residues on protein loops that are inside the lipid membrane than on the outside loops (von Heijne, 1992). On the 4 generalized transmembrane helices, there were 9 positive residues of a total of 17 on the inside portion of the peptide chain including the K-R doublet. On the other hand, of the 9 residues on the outside portion, there were no positive residues. Unfortunately this was not a balanced comparison, because there were not equal numbers of amino acids on both the inside and outside of the lipid membrane. Only 2 generalized transmembrane helices were long enough to represent residues on the outside of the lipid membrane. Nonetheless, the generalized transmembrane helices do not contradict the positive inside rule.

Generalized nontransmembrane sequences

The generalized nontransmembrane sequences contained a region rich in cysteine corresponding to the inside lipid portion of the generalized transmembrane helices. The main body of the nontransmembrane sequences, corresponding to the transmembrane spanning region, contained 17–18 charged residues composed of 50% lysine and arginine, 25% glutamate and aspartate, and 25% glutamine and asparagine. The outside lipid portion also contained charged sequences as well as cysteine residues.

The generalized nontransmembrane sequences should not be considered real peptides in the sense that they represent all other peptides of this length. Rather, the generalized nontransmem-

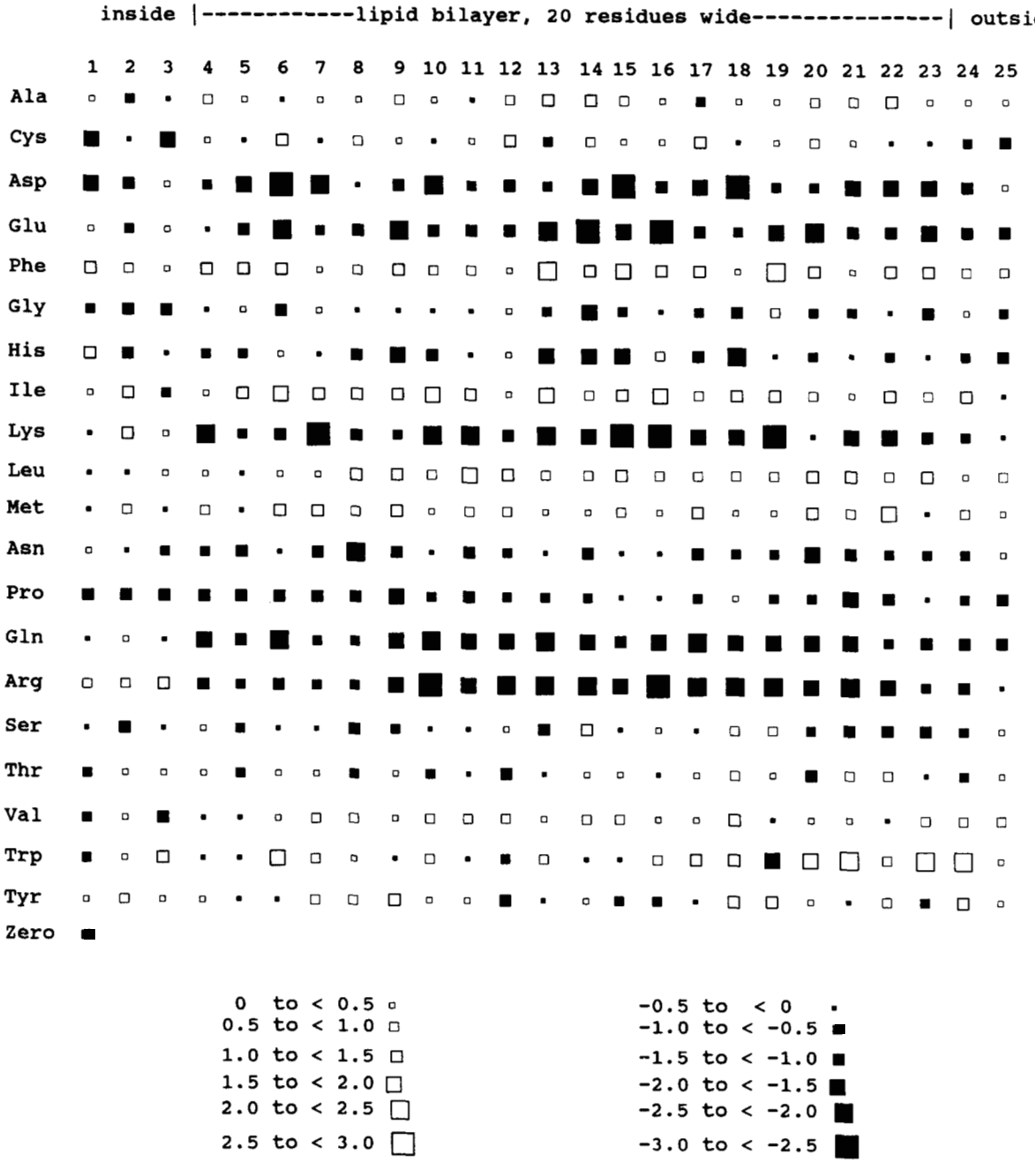

Fig. 3. Hinton diagram of neural network 11 weight matrix. The 20 amino acid types are arranged vertically. The position of each amino acid in the 25-residue window is represented horizontally. Open boxes indicate weights that favor transmembrane helix structure, and closed boxes disfavor that structure.

brane sequences should be thought of as the logical converse of the generalized transmembrane helices. Amino acid residues included in the nontransmembrane sequences are those least likely to be found in the transmembrane helices. The presence of those amino acids in the positions as shown by the nontransmembrane sequences were important landmark residues used by the neural network in order to categorize a peptide as a nontransmembrane sequence.

Proline was not one of the most important residues in distinguishing the generalized nontransmembrane sequences, even though it is thought to break an α-helix by imparting a 30° bend in the otherwise linear helix. The absence of proline in the generalized nontransmembrane sequences is probably due to the fact that proline residues occur naturally in some transmembrane helices (Brandl & Deber, 1986; Williams & Deber, 1991). One model of the G-protein-linked receptors shows proline residues in 4 of the 7 transmembrane helices and suggests that the inclusion of proline in the helix allows for greater conformational responsiveness during ligand binding than does the rigid helix (Liao et al., 1989). Therefore, the neural networks correctly

Table 4. *Generalized transmembrane and nontransmembrane 25 residue sequences as determined from 4 different neural networks*[a]

Network number	Inside |---------- Lipid bilayer, 20 residues wide ----------| Outside
	Transmembrane sequences
	• • • • • • • • • •
8	K W P K F K R F Y H I I I I C L F V L L I M L W M
10	R W Q K I K R F I M W I L I I I F A L F C V F W W
11	H K R F F W I I Y I L L F F F I F Y F W W M W W F
12	Y V L C F L C F F L F T F F W W L W M D I F F T I
	Nontransmembrane sequences
	• • • • • • • • •
8	G G W G D C D K D K K N Q K K Q K D D E E K K K E
10	E D H Q C P V K S D K E R R K D H R D E Q R K E R
11	C G C K D D K N E R K R Q E D R Q D R E R D D Q C
12	D K R K R K R K Q D E E K R E Q K D H C R D T D H

[a] A • indicates at least 3 out of 4 identical amino acids in each sequence at that position.

afforded proline a lower weight in distinguishing the transmembrane helix from the nontransmembrane sequences.

Predictions of transmembrane helices in bacteriorhodopsin

Even though the neural networks reported here were trained for a different purpose (that of discerning the key elements of a transmembrane helix), we attempted to use neural network 10 to predict the placement of the 7 transmembrane helical regions in bacteriorhodopsin as a way of showing the limitations of the training method. The bacteriorhodopsin sequence was written as all possible 25-residue peptides starting with position 1, methionine (M) of the precursor protein. This produced 238 overlapping test sequences that were read in both the N-terminal and then the C-terminal directions. Figure 4 shows the actual positions of the 7 transmembrane regions (Kyte & Doolittle, 1982; von Heijne, 1992). Also shown are 2 sets of transmembrane helix index values plotted as a function of the first amino acid position of the test peptide. The transmembrane helix index has a range of 0–8, where 8 is the most probable transmembrane peptide. In bacteriorhodopsin, like many other G-protein coupled receptors, the first transmembrane helix is inserted with the C-terminal end on the inside of the lipid bilayer. Subsequent helices alternate insertion orientation with the second helix being inserted N-terminal first, and the last inserted C-terminal first. With an ideal prediction system, it would be anticipated that the neural network would correctly find the 7 bacteriorhodopsin transmembrane sequences, that 4 helices (1, 3, 5, and 7) would be found only by reading from the C-terminal direction, and the remaining 3 helices (2, 4, and 6) would be predicted only by reading from the N-terminal direction. In reality, network 10 found a great number of false-positive sequences forming, more or less, a distribution around the actual start of the transmembrane sequences. This overabundance of false-positive sequences is due to the nature of neural network training and not to the multihelical nature of the protein. All our networks were trained with

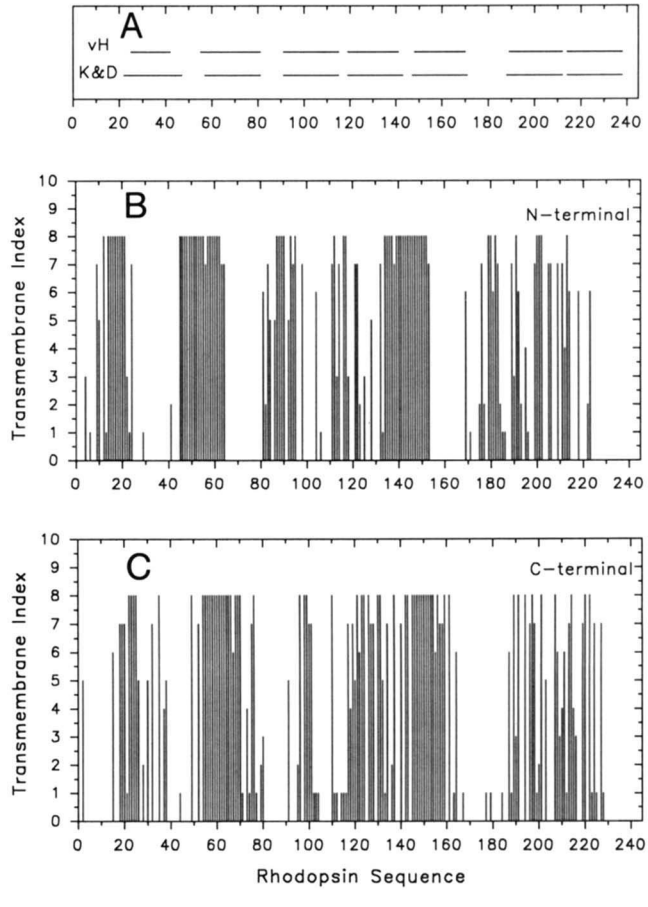

Fig. 4. Prediction of 7 transmembrane helices in bacteriorhodopsin by network 10. **A:** The recognized transmembrane spanning regions in the protein (Kyte & Doolittle, 1982; von Heijne, 1992). Numbering begins with the 13-amino acid precursor. **B:** The determined transmembrane index value of the 238 peptides. Each peptide was taken from the primary structure of bacteriorhodopsin using a sliding window of 25 residues and read from the N-terminus. **C:** The transmembrane index value for the same 238 peptides read in the C-terminal direction.

discrete and nonoverlapping peptide sequences. In this type of training, a sequence of continuous amino acids running from positions $n \ldots n + 25$ would be assigned to the same structural class (HELIX or NON-HELIX) as the displaced, overlapping sequences running from positions $n + 1 \ldots n + 26$ or $n + 2 \ldots n + 27$ or even $n - 1 \ldots n + 24$ or $n - 2 \ldots n + 23$, as well as other overlapping but displaced sequences. In order to make a useful neural network for in situ prediction of the constituent amino acids in the transmembrane helical regions of a protein, our future work will have to include training with sliding windows for centroid amino acid prediction using overlapping peptides from both the transmembrane and nontransmembrane groups.

Conclusions

The main result of this work is that it provides another example of how neural networks are useful to protein scientists interested in analyzing secondary structure. Previously, the use of neural networks had been confined to those scientists with computer backgrounds strong enough to permit them to write their own neural network programs. Now commercial neural network programs are available that have many built-in features.

Our work with proteins containing transmembrane helices shows that even a simple neural network containing only a single hidden neuron is useful in analyzing transmembrane structure. The use of such a neural network leads directly to the creation of "generalized" structures that have heuristic value as model transmembrane helices.

One of the limits of using neural networks for protein structure analysis or prediction is the need for a large training set with as few homologous sequences as possible. This limitation should be overcome in the near future as the size of the international protein databases continues to increase. A second limitation to structure prediction in general is whether any method, including neural networks, based only on local amino acid sequences, can give 100% predictions of secondary structure (Hayward & Collins, 1992). This fundamental problem may require sequence information from not only local but also more distal regions of a protein. This area of research may also be explored by neural networks.

Methods

Database

The database in its final form consisted of 185 transmembrane proteins (928 transmembrane helical sequences) and 131 other proteins (1,018 nontransmembrane sequences) (see Diskette Appendix). All primary sequences were entered into the database as strings of 43 consecutive amino acids. Each amino acid was represented using the standard 1-letter code (e.g., A, alanine; C, cysteine; ... Y, tyrosine). Later in processing the database, these sequences were truncated to 25 amino acids and numbers were added to the letters in order to indicate sequence position (e.g., A1 T2 K3 ... G25).

Transmembrane proteins with known amino acid sequences and known helix orientations were taken from the literature. There were 3 classes of transmembrane proteins. The first class, 49 proteins (26%), consisted of single-pass receptor proteins or membrane-linked binding proteins. The second class, 75 proteins (41%), consisted of 7-pass, G-protein-linked receptors including multiple members of the adrenergic, muscarinic, dopamine, odorant, and opsin receptor groups (see reviews by Findlay & Eliopoulos, 1990; Dohlman et al., 1991). The third class, 61 proteins (33%), consisted of various multipass proteins containing 2–15 spans, including multiple members of the ATPase, ATP synthase, cytochrome *b*, and cytochrome oxidase groups (see review by Rao & Argos, 1986). Transmembrane sequences were oriented in the database from inside to outside the membrane. This meant that about half were written in the usual N-terminal to C-terminal direction and the other half were written in the reverse direction of C-terminal to N-terminal. Once oriented in this manner, the 43-residue, transmembrane sequences were aligned so that the known lipid boundary came between the 12th and 13th amino acid.

There were 131 proteins (1,018 sequences) in the nontransmembrane group. These were taken from the literature (Levitt & Greer, 1977) and from the PIR-International Protein Sequence Database (release 35, January 1993, on *Protein Science* Volume 1 CD-ROM). Functional enzymes, soluble proteins, and precursor proteins were used, as well as the globular regions of single-pass transmembrane proteins (see Diskette Appendix). All nontransmembrane sequences were entered into the database as strings of 43 amino acids written in the usual order of N- to C-terminal. None of these strings were reversed. The names of all proteins used in the complete database are provided on the Diskette Appendix.

On average, every protein in the initial amino acid database (from which both the training set and test set were derived) had approximately 2 additional homologous sequences. In other words, 62% of the initial amino acid database consisted of homologous sequences. In some specific cases, the number of homologous sequences was greater than the average. There were 7 examples of Na-K ATPase and 7 examples of both myoglobin and ribonuclease. In other cases there was only 1 example of a particular protein: epidermal growth factor receptor, MalF transport protein, and papain. In our experiments, the use of homologous sequences was necessary in order to increase the training set size to 2–3× the number of neural connections. Undoubtedly as the size of the protein database grows, it will be possible to achieve the high numbers of sequences necessary for good generalization without resorting to the use of homologous sequences.

Neural networks

Neural networks were created using the commercially available program *BrainMaker*, standard version 2.5, from California Scientific Software, Nevada City, California. This is a general neural net program that allows for creation of an input layer, up to 6 hidden layers, and an output layer. We applied this general program to our database of protein sequences. The standard version has a limit of 512 input neurons. There is a professional version with greater input and total neural capacity.

BrainMaker, as with other neural network programs, creates a trainable, artificial, neural-like network consisting of multiple interconnected units. Because there is a general resemblance to the structure of a biological nervous system, terms are borrowed freely when describing a mathematical neural network

(Bohr et al., 1988). Such a network will produce an output in response to an input. Each unit is called a neuron; it receives input from other connected neurons and passes on an output to different neurons. The gating effect of an individual neuron (whether it fires or not) is determined by the weighted sum of input values multiplied by a continuous transfer function. Different types of transfer functions can be used. There are linear, triangular, and Gaussian functions, but the most common is a sigmoidal transfer function (Lawrence, 1993).

For our purposes, *BrainMaker* was used to create feed-forward, back-propagation networks with a single hidden layer using a sigmoidal transfer function. Feed-forward means that the neurons were arranged in a layered structure with the input neurons on the bottom connected to the single hidden layer, which in turn was connected to the output layer. Back-propagation means that readjustment of the neuron connections (weights) was based on a steepest-descent algorithm according to the output errors (Rumelhart et al., 1986).

The chosen size of our neural networks was a compromise made between the expected size of a transmembrane helical peptide (18–20 amino acids) and the number of allowable input neurons with the standard version of *BrainMaker*. All neural networks had the specific structure of 500 input neurons (20 amino acids × 25 positions), 2 constant firing threshold neurons, and 1 hidden neuron and 2 output neurons labeled HELIX and NON-HELIX. The 2 constant firing, threshold neurons were located, one each, at the input and hidden layers. The addition of threshold neurons is a built-in feature of *BrainMaker* that allows for some output even if the values from all other input neurons are otherwise 0. The training of each successive neural network was begun after more protein sequences were added to the previous sequence database. This was followed by training and testing the new network with no carryover from the previous network. Each new network was independent of the others. This process was repeated 11 times, forming 12 different neural networks. The exception was that network 8 and network 9 were trained on the same database but with differences due to the addition of noise to the data set used for training network 9.

Training sets

Before using the amino acid data set, an adjustment was made so that the original 43-residue sequences were trimmed to a length of 25 amino acids. This was done in order to accommodate the largest possible *BrainMaker* input restriction of 500 neurons (25 long × 20 amino acids). Trimming was done by cutting off a few amino acids from each end of the sequences. In training sets prepared for networks 1–10, amino acid sequences were trimmed so the lipid boundary fell between the seventh and eighth amino acid. This allowed the network to train with some representative residues from inside the lipid boundary even though these sequences were not long enough to penetrate the whole lipid bilayer. In the training set prepared for network 11, sequences were trimmed so the lipid boundary fell between the third and fourth amino acids. Cropping the training set closer to the lipid boundary allowed network 11 to train with sequences that extended completely through the lipid bilayer in most proteins and also contained some representative amino acids from outside the lipid layer. The training set for network 12 was prepared so that the first residue of the 25-amino acid sequences began at the third residue inside the helix portion of the lipid bilayer. This allowed for a 7-member tail on the outside of the lipid layer.

Test sets

BrainMaker software contains a subprogram, *NetMaker*, that converted the amino acid sequences into a training file format for *BrainMaker*. In this process, 10% of the amino acid sequences were reserved in a test set that was subsequently used to determine how well the neural network had generalized information from the training set. Each network was trained and tested with a different data set. The percentage of correct sequence assignments in the test set was recorded as the test set sorting accuracy for each neural network.

BrainMaker software was constructed to train a given network until 100% of the facts in the training set are learned. Training tolerance was preset for each network at the 0.1 level. This meant that when the determined output assignment matched the training pattern to within 10% of the correct assignment, no further adjustments were made to the neural-connection weight matrix (Lawrence, 1993). Testing tolerance was set at the 0.4 level for networks 1–9 and at the 0.2 level for networks 10–12. This meant that if the test sequence pattern matched the training set pattern within 40% (later, within 20%) then the output assignment was made.

Acknowledgments

We are grateful to Drs. Charles Lucas, Anna Ledgerwood, and Jonathon Saxe for their support and the use of their computers.

References

Bohr H, Bohr J, Brunak S, Cotterill RMJ, Lautrup B, Vorskov L, Olsen OH, Petersen SB. 1988. Protein secondary structure and homology by neural networks. The alpha-helices in rhodopsin. *FEBS Lett 241*:223–228.

Brandl CJ, Deber CM. 1986. Hypothesis about the function of membrane-buried proline residues in transport proteins. *Proc Natl Acad Sci USA 83*:917–921.

Dohlman HG, Thorner J, Caron MG, Lefkowitz RJ. 1991. Model systems for the study of seven-transmembrane-segment receptors. *Annu Rev Biochem 60*:653–688.

Dubchak I, Holbrook SR, Kim SH. 1993. Prediction of protein folding class from amino acid composition. *Proteins Struct Funct Genet 16*:79–91.

Findlay J, Eliopoulos E. 1990. Three-dimensional modelling of G protein-linked receptors. *Trends Protein Sci 11*:492–499.

Hartmann E, Rapoport TA, Lodish HF. 1989. Predicting the orientation of eukaryotic membrane-spanning proteins. *Proc Natl Acad Sci USA 86*:5786–5790.

Hayward S, Collins JF. 1992. Limits on α-helix prediction with neural network models. *Proteins Struct Funct Genet 14*:372–381.

Hirst JD, Sternberg MJE. 1992. Prediction of structural and functional features of protein and nucleic acid sequences by artificial neural networks. *Biochemistry 31*:7211–7218.

Holley LH, Karplus M. 1989. Protein secondary structure prediction with a neural network. *Proc Natl Acad Sci USA 86*:152–156.

Jones MK, Anantharamaiah GM, Segrest JP. 1992. Computer programs to identify and classify amphipathic α helical domains. *J Lipid Res 33*:287–296.

Kneller DG, Cohen FE, Langridge R. 1990. Improvements in protein secondary structure prediction by an enhanced neural network. *J Mol Biol 214*:171–182.

Kyte J, Doolittle RF. 1982. A simple method for displaying the hydropathic character of a protein. *J Mol Biol 157*:105–132.

Ladunga I, Czako R, Csabai I, Geszti T. 1991. Improving signal peptide prediction accuracy by simulated neural network. *Comput Applic Biosci 7*:485–487.

Lawrence J. 1993. *Introduction to neural networks, 5th ed.* Nevada City, California: California Scientific Software Press. pp 211, 301–305.

Levitt M, Greer J. 1977. Automatic identification of secondary structure in globular proteins. *J Mol Biol 114*:181–239.
Liao CF, Themmem APN, Joho R, Barberis C, Birnbaumer M, Birnbaumer L. 1989. Molecular cloning and expression of a fifth muscarinic acetylcholine receptor. *J Biol Chem 264*:7328–7337.
McGregor MJ, Flores TP, Sternberg MJ. 1989. Predictions of beta-turns in proteins using neural networks. *Protein Eng 2*:521–526.
Qian N, Sejnowski TJ. 1988. Predicting the secondary structure of globular proteins using neural network models. *J Mol Biol 214*:865–884.
Rao JKM, Argos P. 1986. A conformational preference parameter to predict helices in integral membrane proteins. *Biochim Biophys Acta 869*:197–214.
Reithmeier RAF, Deber CM. 1991. Intrinsic membrane protein structures: Principles and prediction. In: Yeagle P, ed. *The structure of biological membranes*. Boca Raton, Florida: CRC Press. pp 337–393.
Rumelhart DE, Hinton GE, Williams RJ. 1986. Learning representations by back-propagating errors. *Nature 323*:533–536.
Sasagawa F, Tajima K. 1993. Prediction of protein secondary structures by a neural network. *Comput Applic Biosci 5*:147–152.
von Heijne G. 1981. Membrane proteins. The amino acid composition of membrane-penetrating segments. *Eur J Biochem 120*:275–278.
von Heijne G. 1992. Membrane protein structure prediction hydrophobicity analysis and the positive-inside rule. *J Mol Biol 225*:487–494.
Williams KA, Deber CM. 1991. Proline residues in transmembrane helices: Structural or dynamic role. *Biochemistry 30*:8919–8923.